**PART B**

**AIM OF THE PROJECT :** Micro-Project On Double Linked List

**Brief Description :**

* Double Linked List Is Used In Undo Redo Functionality.
* Double Linked List Is Used To Represent A Deck Of Cards In A Game.
* Double Linked List Is Used In Application That Have The Function Of Displaying The Recent Activities.
* The Project Consist Of 4 Main Modules That Are As Follows: -
* Create A Double Linked List, This Creates A New Nod
* Display A Double Linked List; This Displays The Nodes Present IN The List.
* Insert An Element In The List This Option Helps To Insert Element In The Node.
* Delete An Element From The List, It Deletes The Node And The Elements Stored In It.

**Aim of micro project :**

This micro project aims at:

* By using ‘c’ language, application of data structure can be more easily developed.
* Detail study of linked list data structure.
* To perform various operations on linked list.
* Study of new header file that is memory allocation and de-allocation(Malloc.H).

**Course Outcome integrated :**

Implement basic operation on Double Linked List

**Actual procedure followed :**

1. **Group Formation:-** DSU subjects helps us to make a systematic use of data structers in c programming language. The basic aim of micro- project is to accelerate the attainment of the variouse outcome in the course.In the first 2 weeks of the subject July was introduced .The syllabus as well as detail of micro-project was discussed.The group of 3 memebers were formed and the group leaders were selected.The schedule of Plan “A”,”B”& “PRESENTATION” were finalized.The variouse micro-project topics related to subject was discussed our guide gave us the opportunity to select the topic of our choice.
2. **Finalization Of Micro-Project:-** After attending the lectures for 2 weeks.We selected the topic for micro-project.We discussed the topic with our Guide regarding the concept which we are going to apply in the project.We individually tried to explain the basic platform of project.
3. **Planning:-** After finalization of the project we started working on the project.we started the planning phase.We discussed among ourselves regarding the resources such as hardware & software requirements,compiler.In this week we completed ‘PART A PLAN’ of the micro-project which is nothing but a initial description about the project.We submitted it to the guide.
4. **Module Distribution &Analysis Part:-** Once the planning was over regarding resources,etc.We finalized the module which we will be designing.According to members we distributed the module among them.We started the analysis of project.
5. **Design Part :-** In this part we created algorithm & flowchart for our micro-project .By doing this our queries related to project got cleared. With the help of this we were able to explain the guide how our project will actually work.
6. **Implementation :-** In the week we actually started the technical phase .In this phase we technically applied the algorithm & flowchart for each module.The coding for each module was done each member was writing code according to module assighned to them.
7. **Presentation :-** In this week we have to present the micro-project in front of the guide.Each member of group presented their own parts with confidence in front of guide.She asked us variouse queries regarding the topics.We presented the details of each concept of ‘C’ that we used in the project.She asked us to do variouse changes regarding some topics.
8. **Submission:-** This week was submission week.We submitted our project along with ‘Part A & B Plan’ to the guide.We also submitted the hard copies and soft copies of project to the guide

**Actual resources used :**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SR NO | NAME OF RESOURCES USED | SPECIFICATION | QUANTITY | REMARKS |
| 1 | COMPUTER | **PROCESSOR**-AMD  **HARD DISK-**2 TB  **RAM**-16GB  **OPERATING SYSTEM**-WINDOWS 10 PRO | 1  1  1  1 | HARDWARE & SOFTWARE USED |
| 2 | TURBO C | TURBO 4.0 |  | IDE & COMPILER FOR C PROGRAMMING |
| 3 | PROGRAMMING LANGUAGE | C |  | PROGRAMMING LANGUAGE |

**Output of the micro project :**

The output of the project is according to following index :

|  |  |
| --- | --- |
| SR NO | CONTENT |
| 1 | ALGORITHM |
| 2 | FLOWCHART |
| 3 | CODING |
| 4 | CODE OUTPUT |
| 5 | REFERENCES |

FLOW CHART

**start**

**Scanf(“%d&choice”);**

**printf("\n1: Create");**

**printf("\n2: Insert");**

**printf("\n3: Delete");**

**printf("\n4: Display");**

**printf("\n5: Exit");**

**do**

**List(struct code\*malloc(sizeof (struct node)) if->nex=NULL**

**Declare struct node\*list n,pos,choice**

**Switch choice**

**stop**

**while choice!=5**

**Case4**

**Case3**

**Case2**

**Case1**

**void insert(struct node \*list,int n, int pos)**

**{ struct node \*temp=list, \*newnode,\*temp1;**

**int i;**

**newnode=(struct node \*)malloc(sizeof(struct node));**

**newnode->data=n;**

**newnode->next=NULL;**

**for(i=1;(i<pos) && (temp->next!=NULL);i++)**

**temp=temp->next;**

**temp1=temp->next;**

**newnode->next=temp1;**

**temp1->prev=newnode;**

**temp->next=newnode;**

**newnode->prev=temp;**

**}**

**void display(struct node \*list)**

**{**

**struct node \*temp;**

**for(temp=list->next;temp!=NULL;temp=temp->next)**

**printf("%d\t",temp->data);**

**}**

**Enter elements and pos**

**Scanf(“%d”,&n,&pos);**

**void create (struct node \*list)**

**{ int i,n;**

**struct node \*newnode,\*temp=list printf("\nHow many nodes :");**

**scanf("%d",&n);**

**for(i=1;i<=n;i++)**

**{newnode=(struct node \*)malloc(sizeof(struct node));**

**scanf("%d",&newnode->data);**

**newnode->next=NULL;**

**temp->next=newnode;**

**newnode->prev=temp;**

**temp=newnode;**

**} }**

**void detele(struct node\*list,int pos)**

**{**

**struct node \*temp,\*temp1,\*temp2;**

**int i;**

**for(i=1,temp=list;(i<pos) && (temp!=NULL); i++)**

**temp=temp->next;**

**if(temp==NULL)**

**{ printf("Position Out of range");**

**return;**

**}**

**temp1=temp->next;**

**temp2=temp1->next;**

**temp->next=temp2;**

**temp2->prev=temp;**

**free(temp1);**

**}**

**void display(struct node \*list)**

**{ struct node \*temp;**

**for(temp=list->next;temp!=NULL;temp=temp->next)**

**printf("%d\t",temp->data);**

**}**

**Enter elements and pos**

**void display(struct node \*list)**

**{ struct node \*temp;**

**for(temp=list->next;temp!=NULL;temp=temp->next)**

**printf("%d\t",temp->data);**

**}**

**Scanf(“%d”,&n,&pos);**

ALGORITHM

1. **ALGORITHM TO INSERT ELEMENT AT GIVEN POSITION:**

**Step 1:** if free\_loc=null then write overflow goto step 11

**Step 2:** set new\_node=free\_loc

**Step 3:** set free\_loc=free\_loc🡪next

**Step 4:** set new\_node🡪data=val

**Step 5:** set new\_node🡪prev=null

**Step 6:** repeat step 8 while ptr🡪data! =num

**Step 7;** set ptr=ptr🡪next [end of loop]

**Step 8:** new\_node🡪next=ptr🡪next

**Step 9:** set new\_node🡪prev=ptr

**Step 10:** set ptr-🡪next=new\_node

**Step 11:** exit

**B. ALGORITHM TO DELETE NODE**

**Step 1:** if start=null then write underflow go to step 9

**Step 2:** set ptr=start

**Step 3:** repeat step4 while ptr-🡪data! =num

**Step 4:** set ptr=ptr🡪next [end of loop]

**Step 5:** set temp=ptr🡪next

**Step 6:** set ptr🡪next=temp🡪next

**Step 7:** set temp🡪next🡪prev=ptr

**Step 8:** free temp

**Step 9:** exit

**C ALGORITHM TO CREATE A NEW NODE:**

**STEP 1:** IF FREE\_LOC=NULL THEN WRITE OVERFLOW

GOTO STEP 11 [END OF IF]

**STEP 2:** SET NEW\_NODE=FREE\_LOC

**STEP 3:** SET FREE\_LOC=FREE\_LOC🡪NEXT

**STEP 4:** SET NEW\_NODE🡪DATA=VALUE

**STEP 5:** SET NEW\_NODE🡪NEXT=NULL

**STEP 6:** SET PTR=START

**STEP 7:** REPEAT STEP 8 WHILE PTR 🡪NEXT! =NULL

**STEP 8:** SET PTR=PTR🡪NEXT

**STEP 9:** SET PTR🡪 NEXT=NEW\_NODE

**STEP 10:** NEW\_NODE🡪PREV=PTR

**STEP 11:** STOP

CODING

/\*\*\* list is an empty node at the beginning \*\*/

#include<stdio.h>

#include<conio.h>

#include<stdlib.h>

#include<malloc.h>

struct node

{

int data;

clrscr();

struct node \*next,\*prev;

};

void create (struct node \*list)

{

int i,n;

struct node \*newnode,\*temp=list;

printf("\nHow many nodes :");

scanf("%d",&n);

for(i=1;i<=n;i++)

{

newnode=(struct node \*)malloc(sizeof(struct node));

scanf("%d",&newnode->data);

newnode->next=NULL;

temp->next=newnode;

newnode->prev=temp;

temp=newnode;

}

}

void insert(struct node \*list,int n, int pos)

{

struct node \*temp=list, \*newnode,\*temp1;

int i;

newnode=(struct node \*)malloc(sizeof(struct node));

newnode->data=n;

newnode->next=NULL;

for(i=1;(i<pos) && (temp->next!=NULL);i++)

temp=temp->next;

temp1=temp->next;

newnode->next=temp1;

temp1->prev=newnode;

temp->next=newnode;

newnode->prev=temp;

}

void display(struct node \*list)

{

struct node \*temp;

for(temp=list->next;temp!=NULL;temp=temp->next)

printf("%d\t",temp->data);

}

void detele(struct node\*list,int pos)

{

struct node \*temp,\*temp1,\*temp2;

int i;

for(i=1,temp=list;(i<pos) && (temp!=NULL); i++)

temp=temp->next;

if(temp==NULL)

{ printf("Position Out of range");

return;

}

temp1=temp->next;

temp2=temp1->next;

temp->next=temp2;

temp2->prev=temp;

free(temp1);

}

void main()

{

struct node \*list;

int n,pos,choice;

list=(struct node \*)malloc(sizeof(struct node));

list->next=NULL;

do

{

printf("\n1: Create");

printf("\n2: Insert");

printf("\n3: Delete");

printf("\n4: Display");

printf("\n5: Exit");

printf("\n\nEnter your choice :");

scanf("%d",&choice);

switch(choice)

{

case 1: create(list);

break;

case 2:

printf("\nEnter the element and position");

scanf("%d%d",&n,&pos);

insert(list,n,pos);

display(list);

break;

case 3:

printf("\nEnter the position");

scanf("%d",&pos);

delete(list,pos);

display(list);

break;

case 4:

display(list);

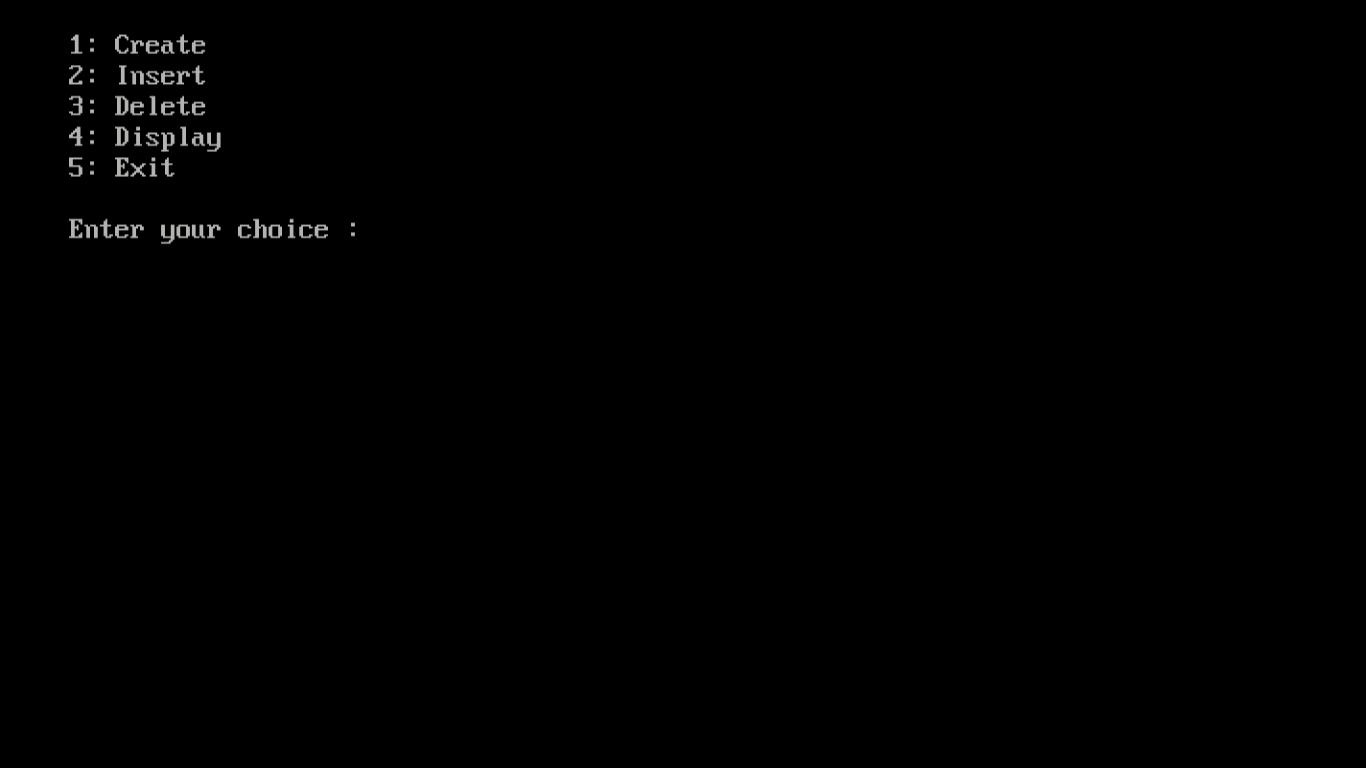
break;

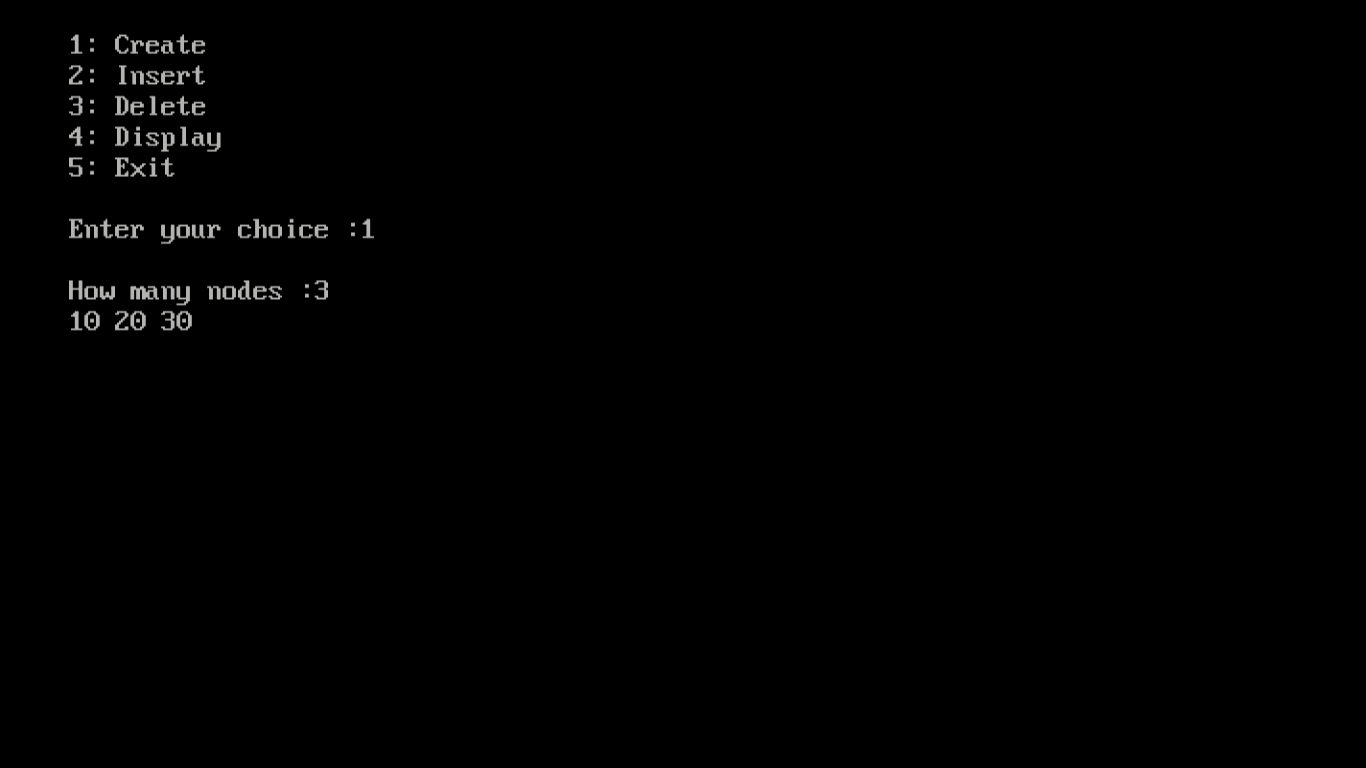
}

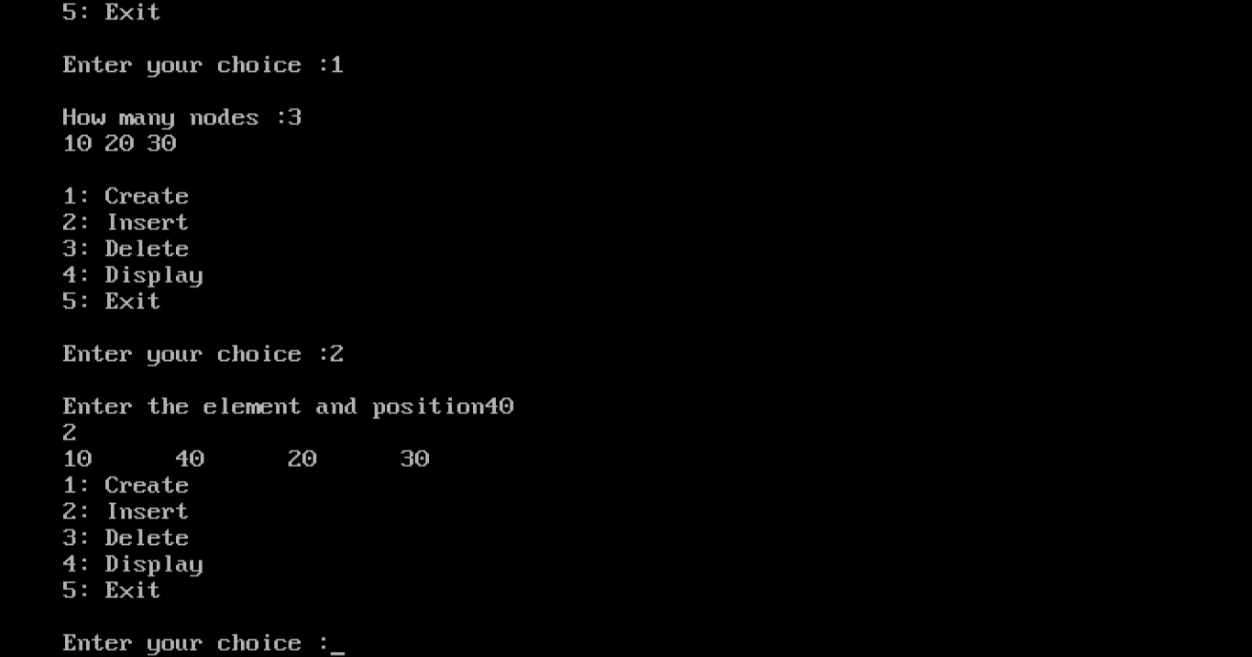
}

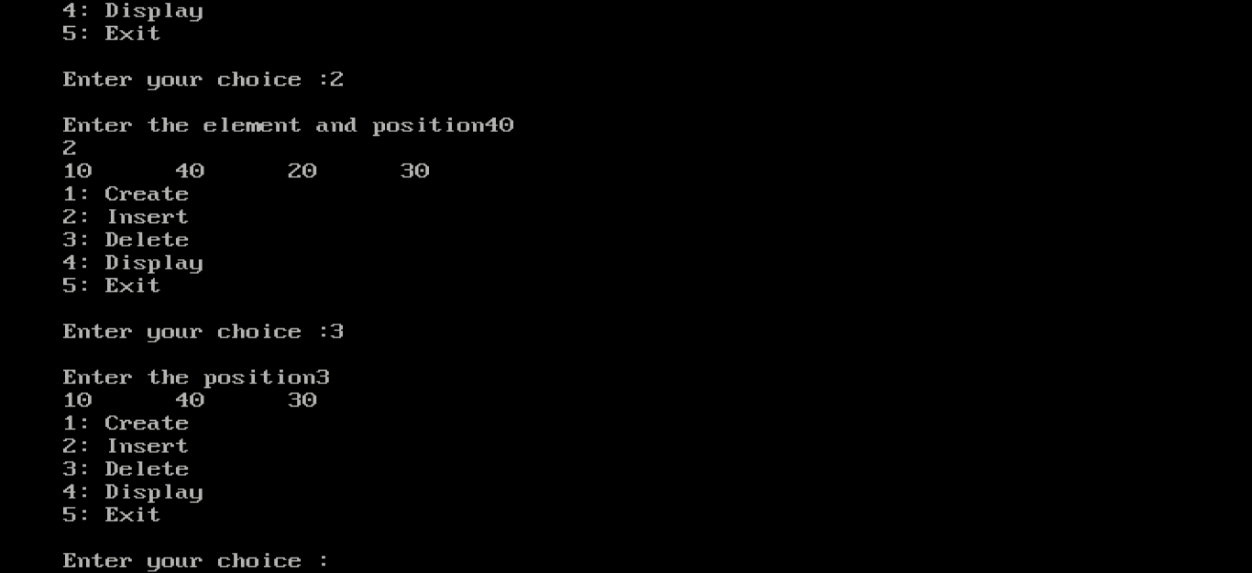
while(choice !=5);

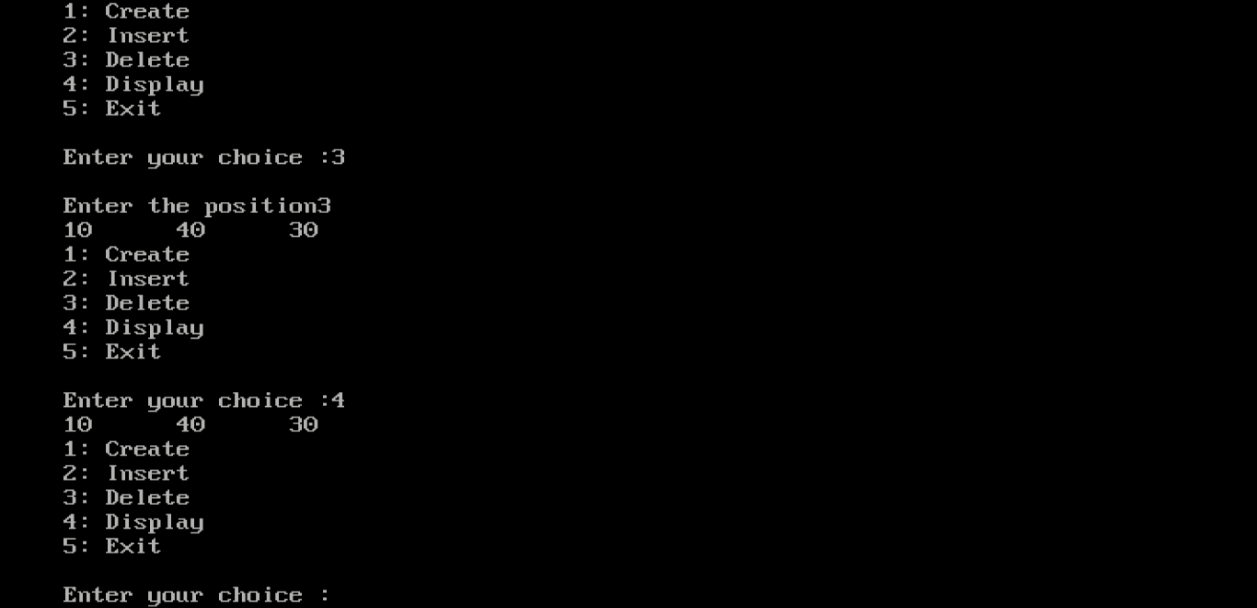
getch();}

CODE OUTPUT









REFERENCES

We do have used a few references during the process of building our project. The references used are from Websites, Books etc.

The references used are:

WEBSITES:

1: www.programiz.com.

2: www.tutorialspoint.com.

3: www.quora.com.

REFERENCE BOOKS:

1: Data structer using ‘c’ by ISRD Group

2: Data structer by Dr Rajendra Kawale

**Skill developed/learning out of this micro project :**

* Since we worked in a group,We developed the skill of ‘TEAMWORK’ in us
* We came to know how and when to use some of the important concept of Data structer in in c programming..
* We attained the satisfied level of programming